

# PATENT ABSTRACTS OF JAPAN

(11)Publication number : **10-303893**

(43)Date of publication of application : **13.11.1998**

(51)Int.Cl.

H04L 12/18

G06F 15/00

H04N 7/15

(21)Application number : **10-089730**

(71)Applicant : **INTERNATL BUSINESS MACH  
CORP <IBM>**

(22)Date of filing : **02.04.1998**

(72)Inventor : **COLYER ADRIAN MARK  
KEY ANDREW  
SETHI VINCENT**

(30)Priority

Priority number : **97 9707316**

Priority date : **10.04.1997**

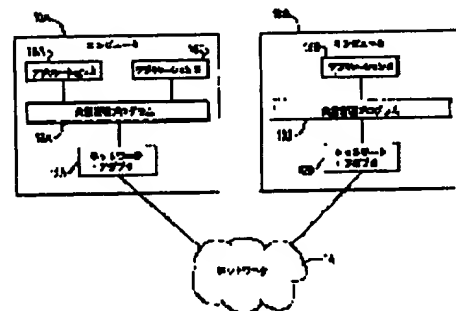
Priority country : **GB**

## (54) PERSONAL CONFERENCE METHOD AND SYSTEM

(57)Abstract:

**PROBLEM TO BE SOLVED:** To secure the equal reference model data among the clients by forming a constitution where every client has the conference model data stored in a server, an instruction is sent to the server when a certain client edits a model and the server immediately processes the received instruction to send the instruction again to the client.

**SOLUTION:** A client computer 32A has a processor 34 which operates by the instructions of client codes 36A, 36B and 36C which are stored in a memory 38 and/or the disk of each client computer. The images are outputted via a video peripheral device 40 of a monitor, etc., connected to the processor 34. The computer 32A stores the current model data 60, receives a model object 36A from the servers 50A to 50D respectively by an apply Op Code method 62 and updates the object 36A. The object 36A is sent to every client.



## LEGAL STATUS

[Date of request for examination] **22.12.1999**

[Date of sending the examiner's decision of rejection] **07.01.2003**

[Kind of final disposal of application other

**Best Available Copy**

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平10-303893

(43) 公開日 平成10年(1998)11月13日

|                           |       |               |
|---------------------------|-------|---------------|
| (51) Int.Cl. <sup>4</sup> | 識別記号  | F I           |
| H 0 4 L 12/18             |       | H 0 4 L 11/18 |
| G 0 6 F 15/00             | 3 9 0 | G 0 6 F 15/00 |
| H 0 4 N 7/15              |       | H 0 4 N 7/15  |

審査請求 未請求 請求項の数17 O L (全 15 頁)

(21) 出願番号 特願平10-89730  
(22) 出願日 平成10年(1998) 4月2日  
(31) 優先権主張番号 9 7 0 7 3 1 6 . 7  
(32) 優先日 1997年4月10日  
(33) 優先権主張国 イギリス (GB)

(71) 出願人 390009531  
インターナショナル・ビジネス・マシー  
ズ・コーポレーション  
INTERNATIONAL BUSIN  
ESS MACHINES CORPO  
RATION  
アメリカ合衆国10504、ニューヨーク州  
アーモンク (番地なし)  
(72) 発明者 エイドリアン・マーク・コルヤー  
イギリス国エスオー19 5イーエス、ハ  
ンプシャー州サウスハンプトン ビターン  
パス・ロード 27  
(74) 代理人 弁理士 坂口 博 (外1名)

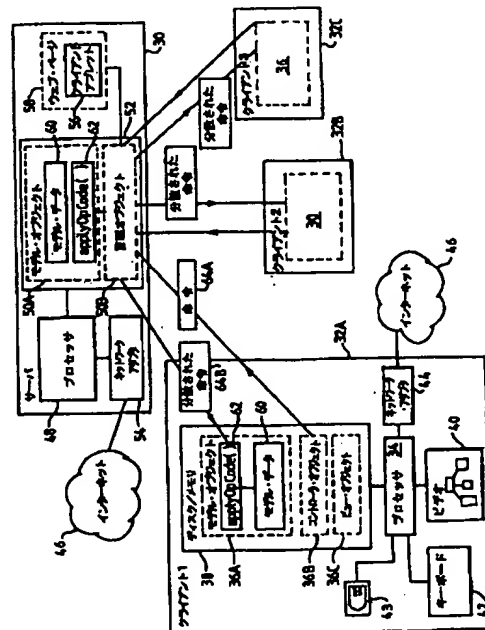
最終頁に続く

(54) 【発明の名称】 パーソナル会議方法及びシステム

(57) 【要約】

【課題】 クライアント／サーバ環境のためのパーソナル  
会議方法およびシステムを提供。

【解決手段】 サーバは共有チョークボードまたは分子モ  
デル等のモデル・データを格納し、各クライアントは会  
議モデル・データのコピーを有する。クライアントの一  
つはモデルを編集してモデルを処理するための命令を生  
成し、該命令をサーバへ送る。サーバはそれ自身の会議  
モデル・データを命令受信と同時に処理し、各クライア  
ントへ再送する。各クライアントは会議モデル・データ  
の受信と同時に同一処理を実行する。それによって、異  
なるクライアントからの異なる処理命令の各々のコピー  
の後に会議モデル・データは等しくなる。



## 【特許請求の範囲】

【請求項1】 接続された一以上のクライアント・コンピュータを有するサーバ・コンピュータのためのパーソナル会議方法であって、

(a) 会議モデル・データをサーバ・コンピュータに格納するステップと、

(b) クライアント・コンピュータからの命令を受信するやいなや前記会議モデル・データを処理するステップと、

(c) 前記サーバ・コンピュータから各々のクライアント・コンピュータに前記命令を再送するステップとを有し、前記会議モデル・データのすべてのコピーが前記サーバ・コンピュータに保持された会議モデル・データに対応するように、各クライアント・コンピュータは該クライアント・コンピュータ上に保持された前記会議データ・モデルのコピーに対して同一の処理を実行するパーソナル会議方法。

【請求項2】 サーバ・コンピュータに接続された一台以上のクライアント・コンピュータのための会議方法であって、

(a) 前記クライアント・コンピュータの各々が会議モデル・データのコピーを格納するステップと、

(b) 前記クライアント・コンピュータの一つが会議モデル・データを処理する命令を生成し、サーバ・コンピュータへ前記命令を送るステップと、

(c) 前記クライアント・コンピュータの各々がサーバ・コンピュータからの命令のコピーを受信し、該命令にもとづいて上記クライアント・コンピュータの各々のモデルを処理するステップとを有し、異なるクライアントからの複数の異なる処理命令の後に、前記会議モデル・データの各々のコピーは互いに対応するパーソナル会議方法。

【請求項3】 前記サーバ・コンピュータは、前記会議モデル・データを要求あり次第クライアント・コンピュータへ前記カンファレンス・モデル・データを送る請求項1または2に記載のパーソナル会議方法。

【請求項4】 上位マシン・バイト・コードの制御下で前記ステップが実行される請求項1ないし3のいずれか一項に記載のパーソナル会議方法。

【請求項5】 クライアント・コンピュータを制御する上位マシン・バイト・コードは、該クライアント・コンピュータで実行されるネットワーク・ブラウザによって前記サーバ・コンピュータからダウンロード可能である請求項4に記載のパーソナル会議方法。

【請求項6】 前記会議モデル・データを処理する前記命令は変数を有するOp codeの形態である請求項2ないし5のいずれか一項に記載のパーソナル会議方法。

【請求項7】 会議モデル・データとクライアント・コンピュータから命令を受信して該命令にもとづいて前記会議モデル・データを処理する手段とを有するサーバ・コ

ンピュータと、命令を生成して該命令を前記サーバ・コンピュータへ送る手段を有する少なくとも2台のクライアント・コンピュータとを備えるパーソナル会議システムであって、

前記サーバ・コンピュータは、

受信した命令をすべてのクライアント・コンピュータへ分散する手段をさらに有し、また前記少なくとも2台のクライアント・コンピュータの各々は、会議モデル・データのコピーと、

前記サーバ・コンピュータからの分散された命令を受信して前記サーバ・コンピュータから受信した命令にもとづいて前記会議モデル・データのコピーを処理する手段とをさらに有するパーソナル会議システム。

【請求項8】 前記少なくとも2台のクライアント・コンピュータの各々は、前記会議モデル・データを表示するための手段をさらに有する請求項7に記載のパーソナル会議システム。

【請求項9】 会議モデル・データと、該会議モデル・データを処理するための命令を生成して該命令を他のコンピュータへ送る手段と、前記命令にもとづいて前記会議モデル・データを処理する手段とを有し、かつメッセージ通信のために接続した少なくとも2台のコンピュータを有するパーソナル会議システムであって前記少なくとも2台のコンピュータの一つがサーバ・コンピュータであり、他のコンピュータが該サーバ・コンピュータと接続されたクライアント・コンピュータであり、前記サーバ・コンピュータは、さらに接続されたクライアント・コンピュータのすべてに分散するための手段を有し、さらに、

前記会議モデル・データを処理する手段は、前記サーバ・コンピュータから分散された命令を受信するための手段を有するもので、

異なるコンピュータからの処理命令のシーケンスは前記会議モデル・データの各々のコピーに対して同時に実行されるパーソナル会議システム。

【請求項10】 サーバ・オブジェクトおよび複数のクライアント・オブジェクトと、モデル・データと命令メッセージに対して前記モデル・データを処理する手段とを有するモデル・オブジェクトとを有し、また前記複数のクライアント・オブジェクトの各々は、前記モデル・オブジェクトを処理するための命令を生成する手段を有するパーソナル会議システムにおいて、

前記複数のクライアント・オブジェクトの各々は関連したモデル・オブジェクトを有し、また前記複数のクライアント・オブジェクトの各々は処理命令をサーバ・オブジェクトに送る手段をさらに有することによって、実行されるべき異なるクライアント・オブジェクトからの命令のシーケンスをモデル・データのインスタンス上の各モデル・オブジェクトによって同時に実行させるパーソナル会議システム。

【請求項11】コンピュータ・ネットワーク全体にわたる伝送のためにモデル・オブジェクトとクライアント・オブジェクトとを実装するための手段をさらに有する請求項10に記載のパーソナル会議システム。

【請求項12】前記モデル・オブジェクトは、現在のモデル・データを含む状態メッセージを発生させ、該状態メッセージを他のオブジェクトに送るために、モデル・オブジェクトを生成する手段をさらに有する請求項11に記載のパーソナル会議システム。

【請求項13】前記モデル・オブジェクトは、前記モデル状態メッセージに送られたものへモデル・データをセットし、それによって一つのモデル・オブジェクトが他のモデル・オブジェクトによってコピーされるために、前記モデル・オブジェクトを生成する手段をさらに有する請求項12に記載のパーソナル会議システム。

【請求項14】さらに、前記サーバ・オブジェクトに関連して集中化されたモデル・オブジェクトを有する請求項10ないし13のいずれか一項に記載のパーソナル会議システム。

【請求項15】クライアント・オブジェクトに関連したモデル・オブジェクトは前記サーバ・オブジェクトに関連した前記モデル・オブジェクトからコピーされる請求項14に記載のパーソナル会議システム。

【請求項16】前記クラスおよびオブジェクトは、上位マシン・バイト・コードにインプリメントされる請求項10ないし15のいずれか一項に記載のパーソナル会議システム。

【請求項17】クライアント・クラスおよびモデル・クラスは、ダウンロード可能で、かつ前記上位マシン・バイト・コードによって使用可能となるネットワーク・ブラウザによって実行可能な上位マシン・バイト・コード・アプレットを含む請求項16に記載のパーソナル会議システム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、パーソナル会議システムに関する。

【0002】

【従来の技術】パーソナル会議システムによって、コンピュータ・ユーザはすべての参加者にとって都合のよい場所に移動するかわりに、ユーザ自身のディスクから会議に参加することが可能となる。電話システムが2加入者以上に対する会議呼出を包含するかたちで発達してきたように、コンピュータ・システムは数多くのネットワーク・コンピュータ・ユーザが一つの会議に参加することを可能とするように発達してきた。パーソナル会議を行う一例として、他のコンピュータ・ユーザとリアル・タイムでテキストまたは音声でミーティングを行うことができるインターネットの「チャット・ルーム」を挙げることができる。「チャット・ルーム」型の会議が発展

すると、パーソナル会議システムが数多くの参加者によるアプリケーションの共有を試みるアプリケーション型会議となる。これの一例として、参加者が「共通」のホワイト・ボードを利用して同一図面の参照および編集が可能な「ホワイト・ボード」アプリケーションが挙げられる。別の例としては、各参加者がコンピュータ・スクリーン上に表示された分子の修飾および処理を行うことができる分子モデリング・アプリケーションが挙げられる。

【0003】いくつかのパーソナル会議システムは、ピア・トゥ・ピア様式でもって実現される。この一例は、IBM Corporation (以下、IBMとする)が開発した製品PM2 PMおよびPerson to Person/2である。この環境下で、例えば図1に示されるように、会議マネジメント/呼出マネージャ・プログラム(18A、18B)はネットワーク(14)全体と通信する1つ以上のネットワーク・アダプタ(12A、12B)のサービスを利用する各コンピュータ(10A、10B)で実行される。いくつかの特別に書かれたパーソナル会議アプリケーション(16A、16B、16C)が会議管理プログラムに接続する。もし会議に共用チョークボードが含まれるならば、例えば、各々の参加しているノード・チョークボード・パーソナル会議アプリケーション(16A、16B、16C)が実行されよう。これらのアプリケーションは互いにデータ・パケットを送ることで、会議マネージャ・プログラム(18A、18B)のサービスが利用できる。

【0004】しかし、このような構成には以下のような欠点がある。

【0005】(i) 会議マネージャ・プログラムは同一の共用チョークボードに対応する各ノード上のアプリケーションのデータ(または他のデータ)が同一であるかどうかを確かめる責任を持たないので、アプリケーションは創出しなければならず、またそのように行うためにそれらの間のプロトコールを使用する。

【0006】(ii) 各ノードは、その対等物が格納するデータと完全に同一なコピーを格納しないことを決める可能性がある。この設計決定は高速実現を促進する傾向にあるので、しばしばすぐには分からない副作用を伴うエラーとなる。

【0007】(iii) もし新規参加者が会議に接続するならば、チョークボード内のデータのコピーはノードの一つから伝達されなければならない。もし各ノード上のデータが同一でなければ、どのデータを送るのが最善か? 転送を効率良く行うために、アプリケーションはネットワーク地理学の実現がさらに複雑化する。新規参加者が会議に接続し、既存のアプリケーションを共有することを望むならば、歩調を合わせなければならない。

【0008】(iv) アプリケーションにおけるこの特別の厄介な問題すべて(歩調合わせを維持するためのピア・トゥ・ピア・プロトコル、転送の接続および切断、地理学の知識)を各種類のアプリケーションで繰り返さなければならない。

【0009】(v) システム内に固有の「会合場所(ミーティング・ポイント)」がないので、人は自分自身で会議に参加するための準備が整っていることを示さなくてもよい。純粋にピア・トゥ・ピア・システムでは、いつでも納得のいく電源投入が期待できるサーバは存在しない。

【0010】いずれのパーティによる共有されたデータの処理を許さないパーソナル会議の一例は、単純化されたピア・トゥ・ピア、例えばData Connection Limitedによって開発されたNeWI環境の共有ホワイトボード・アプリケーションである。ホワイトボードは、一参加者あたり一面が割り当てられる多数の面を有する。各参加者は該参加者自身の面を利用することができ、この面への変更は他の参加者に送られる。したがって、他の参加者は新たな面を見ることができ(しかし、変更を加えることはできない)。誰もが他者の面上のピクチャを変えることができないので、共有データの共有処理はありえない。なお、「NeWI」はData Connection Limitedの登録商標である。

【0011】単純化されたピア・トゥ・ピアはさらに別の欠点を有する。共有ホワイトボードの例では何らかのユーティリティがそのようなツールによって提供されるけれども、ほとんどの値はデータの共有処理にある。このシステムを実現するのは簡単ではあるが、「会議」のこのような様式では共有処理でのテキスト編集および他の形式を実行することが困難である。

【0012】

【発明が解決しようとする課題】したがって、ピア・トゥ・ピア会議システムの主要な問題の一つは、会議モデルの保全性を保つためにすべての宛先/ノードにおいてすべてのデータが同一のオーダに到達しなければならないことである。

【0013】異なる種類のパーソナル会議はIBMによって開発されたXTVシステムによって提供される。このシステムは、Xwindowsシステムの改良/変更である。もし種々の参加者から入力イベント(マウスのクリック、キーストローク等)が所定のマシンに伝わり、実行されているプログラムに適用され、さらに参加者へ返送される画面(またはプログラムのウィンドウ)上に変更が生ずるならば、パーソナル会議の基本形態が達成させる。

【0014】XWindowsはクライアント・サーバ・システムであり、それによってサーバはユーザ入力を提供する機能を果たし、またサーバの画面上に描く機能

を果たす。一般に、各ユーザは該ユーザのマシン上で走るXserver(20)を所有し、またデータ・ストリームはXTVインタセプタを介してXserver(20)に送られX-プロトコルを用いてそれにグラフィック表示させる。Xwindowsがクライアント・サーバであり、またX-プロトコルがそれらの間で転送されることから、異なるマシンで実行されるクライアントおよびサーバにとってそれはありふれたことである。

XWindowsはネットワーキングを実現し、さらにこれは共同作業(パーソナル会議として)が望まれる場合に役に立つ。出力に関しては、XTV(22)はクライアント(24)で実行されているアプリケーションによってサーバ(26)に送られるX-プロトコルを傍受する。それによって、いくつかのX-サーバ(20)へこのデータの出力が広がり、いくつかの参加者に対してプログラム・ウィンドウが見えるようにする。同様に、入力はいくつかのXサーバ(20)から引き出され、共有(多重アクセス可能)アプリケーションが実行されているマシンに振り向けられる。XTVおよびXWindowはIBMの登録商標である。

【0015】標準的なXWindowにおけるクライアントとサーバとのインタラクションを図2に示し、また「中間にある」XTV(22)とのインタラクションを図3に示す。XTV(22)は、いくつかのアプリケーションとサーバとの間で入力および出力トラフィックを転送し、すべての存在する非共同的アプリケーションとともに作業するという利点を有する。

【0016】しかし、図2および図3に示した構成は、以下のような欠点(i)~(v)を有する。

【0017】(i) 共同的アプリケーション「使用可能(イネーブラ)」ではない。

【0018】(ii) 非共同的アプリケーションは一人のユーザを念頭において書かれたものであり、一方キーストロークおよびマウスクリックはいろいろなユーザからのもので、すべてが遅れずに割込する。

【0019】(iii) すべてのユーザが全く同じデータのみを見るだけであり、例えば第二のユーザがノーマル・サイズの表示を見ている一方で、第一のユーザが拡大された表示を見ることができない。

【0020】(iv) アプリケーション・データに対する変更は、数多くの画面の変更(したがって、X-プロトコル)をもたらすことができ、この問題はすべての参加者に対してその時点で送られるという事実によってさらに悪化させる。

【0021】(v) アプリケーションはまさに一つのマシン上でしか実行されず、そのためそれが保持するデータはすべてのマシンが共有ファイル・システム(例えば、AFSまたはDFS)を共有するにもかかわらずそのマシンでしかアクセスすることができない。たとえ、マシンが共通のファイル・システムにアクセスするとし

ても、同一のファイルに対してドライブの文字やパスネームはマシン間で異なるものであってもよい。一般に、遠隔データ処理の場合、遠隔マシン上で実行していることを考慮しなければならない。このことはユーザにとって複雑な物事となる。

【0022】パーソナル会議システムのほとんどの種類に当てはまる別の問題は、新規参加者に対してもっとも近くに存在する（ネットワーク速度の面で）参加者を選択することによってネットワーク・パフォーマンスを最適化する試みがなされていないことである。一般に、パーソナル会議システムはデータを任意の既設ノードから新規ノードへ転送する。

【0023】Hewlett-Packardの欧州特許EP0497022は、分散オブジェクト導引コンピュータ・システムに関するもので、該システムでは共有可能なオブジェクトがクライアント・コンポーネントとサーバ・コンポーネントとに分割される。また、システムはサーバ間でのメッセージの送信およびアップデータの受け渡しを行うために互いに通信する多数のサーバ向けに設計されている。

【0024】

【課題を解決するための手段】本発明の実施形態例では性能に多数の部門を利用する。モデル（Model）／ビュー（View）／コントローラ（Controller）（MVCと略す）の一つは実行中のアプリケーションで使用される一部門である。そのようなアプリケーションは、明確に定義されたインタフェースおよび責務により3つの論理部に分割される。

【0025】モデルは、アプリケーションが表示されて処理可能となるデータである。しかし、モデルそれ自身はデータおよび該データの変更を可能とする入口点（すなわち、方法）のみを含む。モデルは、どのようにしてデータがユーザに対して提示されるかについての概念を持たず、あるいはどのようにして意図するデータの変更／修飾の指示または指定することが許可されるかについての概念を持たない。チョークボード・アプリケーションの例では、モデルは画像を表すビットマップであってもよい。モデルは現在のペンの色、色塗りのスタイル、またはフォント設定等の任意の概念を含むものではない。モデルでのデータの修飾は入力点／方法を介して達成され、さらにモデルの新規の状態は古い状態および入力点（すなわち、入力点は純粋で、汎用データは不使用）に対するパラメータから全体的に決定される。この記述において、入力点／方法はモデルのデータまたは「処理手段」上で処理されることに関連している。

【0026】ビューはユーザに対してモデルの現在の状態を示すためのユーザ・インタフェース・コンポーネントである。それは、非外部または非グローバル変数にアクセスすることで、どのようにして表示設定されたモデル・データから純粋にモデルを表示するかを決定する。す

なわち、特定のモデルを与えることでいつでも同一のビジュアル表示が得られよう。チョークボード・アプリケーションの例では、ビューはウィンドウのクライアント領域にあるモデルのビットマップをスクロールバーとともに表示する。

【0027】コントローラは、ユーザによるモデルに対する変更の指定を可能とさせるユーザ・インタフェース・コンポーネントである。チョークボード・アプリケーションの例では、コントローラはチョークボード上で矩形（または他のグラフィック・シンボル）を描くことをユーザが決定する場合に生ずるラバーバンディングを表示する役割を果たしている。コントローラはモデル（例えば、矩形の描画）を直接修飾することはせず、むしろこの仕事を実行するためにモデルの入力点を呼び出す。コントローラは、ユーザが行ったペンの色およびフォントに対する現在の変更を保つことができ、またこの情報をモデルに送られた矩形描画命令の一部として転送される（変更が完全に指定される目的で）。この記述において、コントローラは処理命令または生成手段の生成することに関連している。

【0028】効率化のためにモデルに変更が加えられた場合、それ自身のビュー（または複数のビュー）にそれが変更された事実を通知する。これによって、変更を反映させるためにビューがそれ自身を描画することが可能となる。あるいは悪影響を受けるそれ自身の部分を選択的に再描画する。

【0029】実際には、コントローラとビューとの違いは、しばしばはっきりとしないものとなる。チョークボードの場合、同一クライアント・ウィンドウはビットマップの現在のビューを表示するとともに、コントローラとの現在のユーザ・インタラクションにもとづいて実施されるラバーバンディングを表示する。

【0030】本発明の実施態様は、これらの問題および他の問題の解決を試みている。

【0031】接続された一台またはそれを上回る数のクライアント・コンピュータ（32A、32B、32C）を有するサーバ・コンピュータ（30）のためのパーソナル会議方法は、（a）会議モデル・データ（60）をサーバ・コンピュータ（30）に格納するステップと、（b）クライアント・コンピュータからの命令（64A）を受信するやいなや会議モデル・データ（60）を処理（62）するステップと、（c）サーバ・コンピュータ（30）から各々のクライアント・コンピュータ（32A、32B、32C）に再送するステップとを有し、各クライアント・コンピュータ（32A、32B、32C）は該クライアント・コンピュータ上に保持された会議データ・モデルのコピーに対して同一を実行する。それによって、会議モデル・データ（60）のすべてのコピーはサーバ・コンピュータに保持された会議モデル・データに対応する。

【0032】クライアントよりはむしろサーバからアップデートされる会議モデルを各クライアントが持つ会議システムのためのクライアント／サーバ構成を用いることは、あきらかに有利である。任意の（非ブロック）パーソナル会議システム設計にとって必須の連番（シリアルゼーション）機構が提供される。すべての編集メッセージは、クライアント・コンピュータに送られる際にサーバ・コンピュータを通過するので、中央にシリアルゼーションを行う場所が提供される。サーバ・コンピュータは、すべてのクライアント・コンピュータに対するデータの同時転送を約束することができ、すべてのクライアント・コンピュータ・システムが同一のものを表示することが保証される。

【0033】さらなる利点は明らかである。例えば、パーソナル会議システムの設計はモデル／ビュー／コントロール構成を取り込むことによって明確化され、かつ単純化される。また、会議モデルに対する変更をインターセプトすることおよびクライアント・コンピュータに対する変更を分散させることは、対応する画面のビューを実現するもっとも単純な方法である。そのような設計において、サーバ・コンピュータはいかなるユーザ・インタフェース・コードからも解放されている。

【0034】サーバ・コンピュータそれ自身は、永久的な「会合場所」を提供する。この「会合場所」でクライアントは該クライアントの存在および共有された作業への参加意志とを宣伝する。このような利点は、ピア・トゥ・ピア・システムでは特に欠けている。

【0035】本発明の第2の態様にもとづくと、サーバ・コンピュータに接続された一台またはそれを上回る数のクライアント・コンピュータのためのパーソナル会議方法を提供するもので、該方法は、(a) 上記クライアント・コンピュータの各々が会議モデル・データのコピーを格納するステップと、(b) 上記クライアント・コンピュータの一つが会議モデル・データを処理する命令を生成し、サーバ・コンピュータへ上記命令を送るステップと、上記クライアント・コンピュータの各々がサーバ・コンピュータからの命令のコピーを受信し、該命令にもとづいて上記クライアント・コンピュータの各々のモデルを処理するステップとを有するもので、さらに、異なるクライアントからの複数の異なる処理命令後に、会議モデル・データの各々のコピーは互いに対応するものとなる。

【0036】本発明の第3の態様にもとづいて、サーバ・コンピュータと少なくとも2台のクライアント・コンピュータとを有するパーソナル会議システムを提供する。このシステムにおいて、サーバ・コンピュータは、会議モデル・データと、クライアント・コンピュータから命令を受信し、該命令にもとづいて会議モデル・データを処理する手段とを備え、また、クライアント・コンピュータは、命令を生成し、該命令をサーバ・コンピュ

ータへ送る手段を備え、さらにサーバ・コンピュータが受信した命令をすべてのクライアント・コンピュータへ分散する手段を有し、またクライアント・コンピュータの各々は、さらに会議モデル・データのコピーと、サーバ・コンピュータからの分散された命令を受信し、サーバ・コンピュータから受信した命令にもとづいて会議モデル・データのコピーを処理する手段とを備えることを特徴とする。

【0037】本発明の第4の態様にもとづいて、メッセージ通信のために接続した少なくとも2台のコンピュータを有するパーソナル会議システムを提供する。このシステムにおいて、各コンピュータは、会議モデル・データと、会議モデル・データを処理するための命令を生成し、該命令を他のコンピュータへ送る手段と、命令にもとづいて会議モデル・データを処理する手段とを備え、さらにコンピュータの一つがサーバ・コンピュータであり、他のコンピュータが該サーバ・コンピュータと接続されたクライアント・コンピュータであり、サーバ・コンピュータは、さらに接続されたクライアント・コンピュータのすべてに分散するための手段を有し、また処理のための手段は、サーバ・コンピュータから分散された命令を受信するための手段を有する。したがって、異なるコンピュータからの処理命令のシーケンスは会議モデル・データの各々のコピー上で同時に実行されよう。

【0038】パーソナル会議システムは、サーバ・オブジェクトおよび複数のクライアント・オブジェクトと、モデル・データ(60)および命令メッセージ(64)に対するモデル・データ(60)を処理する手段(62)を有するモデル・オブジェクト(36A)と、モデル・オブジェクト(36A)を処理するための命令を生成する手段を有する各クライアント・オブジェクト(36)とを備え、さらに、各クライアント・オブジェクト(36)は関連したモデル・オブジェクトを有し、各クライアント・オブジェクト(36)はさらに処理命令をサーバ・オブジェクトに送る手段(36B)を有し、それによって、実行されるべき異なるクライアント・オブジェクト(36)からの命令のシーケンスをモデル・データ(60)の各インスタンス上の各モデル・オブジェクトによって同時に実行することが可能となる。

【0039】好ましくは命令は変数を有するOp codeの形態にある。これは会議モデル・データに対する修飾を実装する上で簡便な方法である。修飾はそれを生成したクライアント・コンピュータからサーバ・コンピュータへ、またサーバ・コンピュータから会議に従事するすべてのクライアント・コンピュータへ簡単に分散されることが可能である。明らかに、もし同一の初期会議モデル・データが多くのクライアント・コンピュータに存在するものならば、同一の処理を会議モデル・データのすべてのコピーに適用することで同一モデルが生ずるであろう。

【0040】好ましくは、会議に参加することを望み、またサーバ・コンピュータに対してそのような要求を行うクライアント・コンピュータへ会議モデル・データのコピーを送る。これによって、新規のクライアント・コンピュータが任意の場所で会議に参加することを可能とさせ、またその場所で正確かつ信頼性のある会議録のコピーを受け取ることを可能とさせる。好ましい実施形態例では、ブランク・テンプレート会議モデルはクライアント・コンピュータ上に格納され、その状態はサーバに保持された会議モデル・データの状態に一致するように変更される。

【0041】好ましい実施形態例では、サーバ・コンピュータおよびクライアント・コンピュータで実行される方法は、Java等の上位バイト・コードの制御下にある。これによって、会議の異機種間の移植性が可能となる。マシン上位バイト・コードはクライアント・コンピュータ上で走るネットワーク・ブラウザによってサーバ・コンピュータからダウンロード可能である。このことは、事前にクライアント・コードをインストールしておく必要性がまったくなくワールド・ワイド・ウェブからリアル・タイムでダウンロードすることですでにインターネットに接続されているほとんどのコンピュータの連結性を容易にする。

【0042】好ましい実施形態では、各クライアント・コンピュータはコンピュータの画面上に会議モデルをグラフィック表示する。したがって、各クライアント・コンピュータは会議モデル・データがそれ自身の画面上、すなわちビューに表示される方法の細目に適応し、一方で同時に他のクライアント・コンピュータの各々と同様の会議モデル・データを表示する。

【0043】好ましい実施形態例では、サーバ・コンピュータは会議モデル・データをグラフィック表示する必要性はないけれどもクライアント・コンピュータ上の会議モデル・データのすべてのコピーを保全する責任を持つことで会議での中心的役割を担う。サーバ・コンピュータ上に集中化された会議モデル・データは、マスタとしての役割を果し、すべてのクライアント・コンピュータに対して歩調を合わせるように強いる。

【0044】上位マシン・コードはJava仮想マシンに必要とされるバイト・コード等のプラットフォームから独立したコードである。Javaバイト・コード等の上位マシン・コードは、アプリケーションから独立したオペレーティング・システム／プラットフォームを提供する。また、Javaバイト・コードは再コンパイルすることなしに任意のJava使用可能プラットフォーム（Java実行可能環境を有するオペレーティング・システムまたは仮想マシン）上で実行可能である。JavaはSUN MICROSYSTEMS CORPORATIONの登録商標である。

【0045】

【発明の実施の形態】本発明の実施形態例であるクライアント／サーバの構成は、図4に示すように、3台のクライアント・コンピュータ（32A、32B、32C）に直接接続したセントラル・サーバ・コンピュータ（30）を有する。クライアント・コンピュータ（32A）は、メモリ（38）および（または）クライアント・コンピュータのディスク上に格納されたクライアント・コード（36A、36B、36C）の命令に従って稼働するプロセッサ（34）を有する。クライアント・コンピュータ（32A）の映像出力は、プロセッサ（42）に接続されたモニタまたはLCD画面等のビデオ周辺機器（40）を介して行われ、またクライアント・コンピュータ（32A）に対するユーザ入力プロセッサ（34）に接続されたキーボード（42）またはコンピュータ・マウス（43）によって行われる。クライアント・コンピュータは、さらにネットワーク・アダプタ（44）を備える。このネットワーク・アダプタ（44）の一端はプロセッサに接続し、その他端はコンピュータ・ネットワーク（46）に接続している。これら2種類のコンピュータ・ネットワークの例として、企業内イントラネットとインターネットとが挙げられる。クライアント・コンピュータの一例は133MHzの486マイクロ・プロセッサ、1ギガバイトのハードディスク、16メガバイトのメモリ、および16インチのモニタを備えたIBMのAptivaコンピュータである。

【0046】サーバ・コンピュータ（30）は、該サーバ・コンピュータのメモリ（52）および（または）ディスク上に格納されたサーバ・コード（50A、50B）（サーバ・プログラム・コード）の命令に従って稼働するプロセッサ（47）を有する。映像出力は、必ずしも必要としないがプロセッサ（48）に接続されたモニタまたはLCD画面等のビデオ周辺機器を介して行うこともできる。また、ユーザ入力も必ずしも必要としないがプロセッサ（48）に接続されたキーボードまたはコンピュータ・マウスを介して行うことができよう。サーバ・コンピュータ（30）は、さらにネットワーク・アダプタ（54）を備える。このネットワーク・アダプタ（54）の一端はプロセッサ（48）に接続し、その他端はコンピュータ・ネットワーク（46）に接続している。これら2種類のコンピュータ・ネットワークの例として、企業内イントラネットとインターネットとが挙げられる。クライアント・コンピュータの一例はIntel P200MHzマイクロ・プロセッサ、20ギガバイトのハードディスク、および64メガバイトのメモリを備えたIBMのRISC6000コンピュータである。

【0047】クライアント・コード（36A、36B、36C）の各々の具体例は、3つの主要部分から構成される。すなわち、会議モデル・オブジェクト（36A）

ラスのインスタンス)、会議コントローラ・オブジェクト (36B) (ConferencingAppletControllerクラスのインスタンス)、および会議ビュー・オブジェクト (36C) (ConferencingAppletViewクラスのインスタンス) である。クライアント・コードは、サーバに格納されたワールド・ワイド・ウェブ (WWW) のページ (58) に含まれるクライアント・アプレット (56) を形成する。クライアント・コード (36) の数多くのインスタンスが存在するのと同じぐらいクライアント・コンピュータ (32A、32B、32C) が存在し、クライアント・コードのインスタンスがダウンロードされ、各クライアント・コンピュータ (32A、32B、32C) に格納される。

【0048】クライアント・コンピュータ (32A) 上にクライアント・コードのインスタンスをダウンロードまたは生成するために、インターネットに接続されたJava使用可能ウェブ・ブラウザを有するコンピュータを必要とする。すなわち、クライアント・アプレット (56) を含む会議ウェブ・ページ (58) にアクセスし、一覧することで、ブラウザはクライアント・コンピュータ (32A) 上にクライアント・アプレット (56) をコピーする。アプレット (56) は普通ダウンロードで動作するが、カーソルおよびマウス (43) を用いてモニタ (40) 上に表示されたボタン・アイコンの選択等のイベント (事象) を行った後も動作することも可能である。

【0049】サーバ・コード (50A、50B) は2つの主要な部分から構成される。すなわち、会議マネージャ (50A) (会議システム・マネージャ・クラスのインスタンス) および会議モデル・オブジェクト (50B) (ConferencingAppletModelクラスのインスタンス) である。サーバ・コードに使用される他のサーバ・オブジェクトは、ServerMainオブジェクト (50C)、Connectionオブジェクト (50D)、およびServerMonitorオブジェクト (50E) である (図4を参照)。

【0050】ServerMainオブジェクト (50C) は、クライアントから入ってくるネットワーク接続を傾聴して受け付ける。接続が受信されると、それによってConnectionクラス (50D) のインスタンスが生成される。ServerMainオブジェクトはまた、たった一つの会議システム・マネージャ・オブジェクトを生成し、かつ所有する。

【0051】Connectionクラスのインスタンス (50B) は、各クライアントに対してConnectionオブジェクトが存在するように、ネットワーク接続がサーバによって受信されるたびに生成される。それによって個々のクライアントに対する接続が表現される。Connectionオブジェクト (50D) は、

クライアントから入ってくるメッセージを続けて傾聴し、該メッセージを会議システム・マネージャ・クラスへ送る。また、Connectionオブジェクト (50D) はConferencingSystemManagerオブジェクト (50B) からクライアントへメッセージを送り返す。

【0052】ConferencingSystemManagerオブジェクト (50B) は、サーバに対して同時に起こるマルチ・パーティ会議を管理する。進行中の各会議は、ConferencingAppletModelクラス (または詳細にはそのサブクラスのうちの一つ) のインスタンスによって表される。進行中の各会議はすべてのメッセージで承認される「モデルID」によってそれ自身のモデル・クラスを識別する。

【0053】ServerMonitorオブジェクト (50E) は、ConferencingSystemManagerオブジェクト (50B) のアクティビティをグラフィックで表現する。このグラフィックス表示は管理者によって会議システムでのアクティビティを監視する際に利用される。

【0054】ConferencingAppletModelクラス (50A) は、すべての会議モデルに対してスーパークラスである。各々の種類のマルチ・パーティ会議はそれ自身の会議アプレット・モデルのサブクラスを生成する。例えば、チャット・アプリケーションのチャット会議モデル、チョークボード・アプリケーションのChalkboardConferenceModel、その他を挙げることができよう。モデル・オブジェクトは、モデル・データ (60) の形で会議の現在の状況を維持する。それによって、以下の3つの重要なメソッドが提供される。

【0055】applyOpCode () (62) は、オペランドおよびそれ自身の変数 (クライアントからのメッセージで受け取られるのと同様に) をモデルに適用し、それによって更新されるモデルのデータ (60) (状態) が生ずるであろう。

【0056】setState () (60) (状態) は、モデルにそれ自身のデータ (60) (状態) をメッセージに送られたものに設定させる。

【0057】getState () は、モデルにそれ自身の現在の状態が含まれるメッセージを生成させる。

【0058】クライアント・コードで使用される他のクライアント・オブジェクトは、Sharerオブジェクト (36D) およびSharerReaderオブジェクト (36E) である (図5参照)。

【0059】ConferencingAppletModelオブジェクト (36A) はサーバで使用されるオブジェクトと同一であり、以下に記述するサブクラスを有する。

【0060】ConferencingAppletV

15

iewクラスはすべての会議ビュー・コントローラ（ユーザに対して会議の現在の状態を表示することに使用、例えばチャット会議の追記）のスーパークラスである。各々の種類の会議はそれ自身のConferencingAppletViewのサブクラスを導入する。クラス・インスタンス（36C）のビューはモデルと一対一に対応する。モデルのデータ（60）が変更されると、それ自身の再描画をさせるビューのinvalidate（）メソッドを呼出す。

【0061】ConferencingAppletView 10  
iewクラスはすべての会議ビュー・コントローラ（サーバへのメッセージ送信およびサーバからのメッセージ読取りに使用）のスーパークラスであり、コントローラ・オブジェクト（36B）を形成する。コントローラ・オブジェクト（36B）はビューおよびモデルと一対一の関係にある。ユーザ・アクションがエンド・ユーザ・インタフェースで実行されると、ビューは実行されるべきアクションのコントローラに知らせる。コントローラは、これをメッセージの中に構成してサーバ（30）へ送る。サーバ（30）はメッセージを会議のすべてのコントローラへ同報通信する。同様に、コントローラは入ってくるサーバからのメッセージを傾聴し、それを評価のためにモデルのapplyOpCode（）メソッド（62）へ送る。 20

【0062】Sharerオブジェクト（36D）は、サーバとのソケット接続を確立するために、そしてメッセージをそれに送るために会議アプレット・コントローラによって使用される。

【0063】SharerReaderオブジェクト（36E）は入ってくるサーバからのメッセージ（30）を傾聴し、それを評価のためにモデルへ送る。 30

【0064】実施形態例の初期化は以下の通りである。WWWブラウザの制御下にあるクライアント・コンピュータ（32A）はクライアント・アプレット（50）が含まれるページ（58）を表示するよう命令される。ページはある種の会議への参加を望むユーザのための会合場所であってもよい。閲覧時、クライアント・アプレット（56）がサーバ（30）からクライアント・コンピュータにダウンロードされ、実行時にクライアント・コードのインスタンスが生成される。このクライアント・コードのインスタンスには、会議アプレット・モデル・オブジェクト（36A）、会議アプレット・コントローラ・オブジェクト（36B）、会議アプレット・ビュー・オブジェクト（36C）、Sharerオブジェクト（36D）、およびSharerReaderオブジェクト（36E）が含まれる。さらに、Sharerオブジェクト（36D）の制御下にあるクライアント・コンピュータ（36A）はサーバ・オブジェクトとソケット接続の確立を試みる。 40

【0065】特定の会議に対するサーバ・コード（50 50

16

A～50E）は、第1のクライアントが存在しない会議への接続を要求するとただちに生成される。次のクライアント・コンピュータがソケット接続の確立を試みと、サーバメイン・オブジェクト（50C）はコネクション・クラス（50D）のインスタンスを生成し、それをクライアント・コンピュータに対応付ける。ServerMainオブジェクト（50C）は、（状態）を含むメッセージを生成するゲート・ステート（状態）を呼び出すことによって会議アプレット・モデル・オブジェクト（50A）にあるモデル・データ（60）の状態を決定する。このメッセージは、ネットワーク接続を介してサーバのコネクション・オブジェクト（50D）によってShareReaderオブジェクト（36E）によってそれがピック・アップされるクライアント・オブジェクト（36E）に送られる。次にメッセージは、そのモデル・データ（60）の状態をそのセット・ステート（状態）方法を用いてメッセージに送られるものに変えるクライアントの会議アプレット・モデル・オブジェクト（36A）に送られる。

【0066】ひとたびクライアント・コンピュータが現在のモデル・データ（60）を格納すると、該コンピュータは演算コード（OpCode）を含む更新・メッセージの一定のフローをapplyOpCode（OpCode）メソッド（62）を用いてそれ自身を更新するために会議モデル・オブジェクト（36A）を要求するサーバ（50A～50D）から受け取る。アップデート・メッセージは会議が進行しているかぎり続く。実際のところ、特定の会議について接続された各々のクライアントに対して、サーバから同一のアップデート・メッセージ（OpCode）が送られる。このようにして、もし会議モデル・オブジェクトのすべてが開始時に同一のモデルを有し、また順序を同じくして同一のメッセージ・アップデート（OpCode）を受け取るならば、会議の任意の場所において、任意の会議モデル・オブジェクトのモデル・データ（60）が一致する。

【0067】もしクライアントがモデル・データ（60）の追加または編集による会議への参加を望むならば、モデル・データ（60）に対する変更は会議コントローラ・オブジェクト（36B）によってオペレーション・コードとして表現され、サーバ・コンピュータ（30）の会議マネージャ・オブジェクト（50B）に送られる。つぎに、会議マネージャ・オブジェクト（50B）は参加しているクライアント・オブジェクト（36）のすべてに対してアップデート・メッセージとしてオペレーション・コードを分配または再送する。各会議モデル・オブジェクト（36A）は更新・メッセージを受け取り、また各applyOpCodeメソッド（62）は各々のモデル・データ（60）の組を更新する。

【0068】サーバ（30）の会議モデル・オブジェクト（50A）もまたアップデート・メッセージを受け取

り、applyOpCode() メソッド(62) はサーバ・オブジェクトのモデル・データ(60)を更新する。

【0069】本発明の一実施態様は、対話型共同作業環境としての企業イントラネットのために開発されたIBMのTeamWorkerアプリケーションである。モデル・データ(60)は、多色で書かれたり、描かれたり、あるいは塗られたりする2次元表面を表すチョークボード・アプリケーションで使用されるデータである。図5は、TeamWorkerのメイン・クラス間のインタラクションを示す。TeamWorkerの他のクラスは、TeamWorkerのインフラストラクチャの上に特定の会議アプリケーションを実装し、それらの会議を開始することができるグラフィカル・フロント・エンドを提供するために使用される。

【0070】ブラウザにダウンロードされる場合にメイン・アプレット・クラス(56)が実行される。それによって会議機能をユーザに表示するために使用されるTeamWorkerwindowが生成される(図6)。TeamWorkerは「チーム」という概念を有し、該チームの各々のメンバーはアプレット内部のTeamMemberModelクラスのインスタンスによって表示される。どのTeamMemberModelも、対応するTeamMemberViewを有する。このTeamMemberViewは、パスポートの写真の大きさのウィンドウにチームの構成員の写真を表示することによってメイン・ウィンドウ内にそのチーム・メンバーを表示する。チーム構成員の写真の左側に、ユーザによる会議機能の開始を可能とさせるツールバーが表示される。使用可能な各会議機能はツールバー内のボタンによって表示される。それらのボタンは、クラス・チャット・ボタン(classes ChatButton) (マルチ・パーティ・チャット会議を開始あるいは該会議へ参加する際に使用)、ダイアリ・ボタン(DiaryButton) (チーム構成員のダイアリを表示する際に使用)、eメール・ボタン(EmailButton) (チーム構成員へeメールを送る際に使用)、ホーム・ページ・ボタン(HomePageButton) (チーム・ワーカ・アプレットが動作するウェブ・ブラウザにチーム・メンバーのホーム・ページの表示を要求する際に使用)、メッセージ・ボタン(MessageButton) (受信側のディスプレイにポップ・アップされるダイレクト・メッセージを他のチーム構成員へ送信する際に使用)、電話ボタン(PhoneButton) (他のチーム・メンバーの電話番号を表示する際に使用)、およびジョグル・ボタン(JoggleButton) (ジョグルと呼ばれるマルチ・パーティ・ゲームを起動する際に使用)によって実現される。

【0071】これらの機能のうちのいくつかは、純粋に

クライアントによって実装され(例えば電話番号およびホーム・ページの表示)、他のものはアプリケーションの主体であるチーム・ワーカ・インフラストラクチャを要求する真のマルチ・パーティ会議アプリケーションである。

【0072】会議ボタンの一つ(例えば、チャット・ボタン)が押された場合、会議を管理するために新規の会議ビュー(36C)、モデル(36A)、およびコントローラ(36D)のインスタンスが生成される。生成された実際のクラスは、会議アプレット・モデル、ビュー、およびコントローラ・クラスのサブクラスであって、これらは所定のタイプの会議に特有なものである。これらのクラスは既に述べたようにして、また既に図示したようにしてチャットを管理するのに使用される。

【0073】本発明は、チョークボード・アプリケーションに限定されるものではない。例えば、本発明の別の実施形態例として化学者および生物学者によって使用されるような3次元分子モデリング・アプリケーションが挙げられ、該アプリケーションにおいて本発明が適用されることが特に有益である。モデル・データ(60)は分子中の種々の原始の種類、場所、および結合を表すものとして使用することができる。会議ビュー・オブジェクトは任意の方向から分子を見ることができよう。また、会議コントローラ・オブジェクトはユーザによる原子の処理および変更を可能とさせる。そのようなアプリケーションは、特に解像度の高い画像を使用することからプロセッサ・インテンシブであり、モデル・データをクライアント・コンピュータに格納することは主な実用的問題である。このことは、アプリケーションが対処する明細書の序文で概説された問題となる。

【0074】プログラミングの性質上、オブジェクト指向プログラミング言語以外のプログラミング言語を用いて本発明を実現することはまったく可能である。

【0075】オブジェクト指向環境下では、一台のクライアント・コンピュータが複数の会議に参加することは可能である。

【0076】もちろん、サーバ・コンピュータは実施例で述べたような3台のクライアント・コンピュータのみならず任意の数のクライアント・コンピュータに実際に接続することができる。さらに、実施例で示したコンピュータを使用する必然性はなく、本発明は異なる構成で接続された別のタイプのコンピュータを用いて実施してもよい。

【0077】以上、クライアント/サーバ環境のためのパーソナル会議方法およびシステムについて説明した。サーバは、共有チョークボードまたは分子モデル等の会議モデル・データ(60)を格納し、また各クライアントは会議モデル・データ(60)を持つ。前記クライアントの一つがモデルを編集する場合、モデルを処理するための命令を生成し、該命令をサーバへ送る。サーバは

命令の受け取り次第会議モデル・データの処理(62)を実行し、命令を各々のクライアントへ再送する。そして、各クライアントは同一処理(62)を各々のコピーされた会議データ・モデル(60)上で実行する。したがって、異なるクライアントからの複数の異なるオペレーティング命令の後に、会議モデル・データ(60)は等しいものとなる。

【0078】まとめとして、本発明の構成に関して以下の事項を開示する。

(1) 接続された一以上のクライアント・コンピュータを有するサーバ・コンピュータのためのパーソナル会議方法であって、(a) 会議モデル・データをサーバ・コンピュータに格納するステップと、(b) クライアント・コンピュータからの命令を受信するやいなや前記会議モデル・データを処理するステップと、(c) 前記サーバ・コンピュータから各々のクライアント・コンピュータに前記命令を再送するステップとを有し、前記会議モデル・データのすべてのコピーが前記サーバ・コンピュータに保持された会議モデル・データに対応するように、各クライアント・コンピュータは該クライアント・コンピュータ上に保持された前記会議データ・モデルのコピーに対して同一の処理を実行するパーソナル会議方法。

(2) サーバ・コンピュータに接続された一台以上のクライアント・コンピュータのための会議方法であって、

(a) 前記クライアント・コンピュータの各々が会議モデル・データのコピーを格納するステップと、(b) 前記クライアント・コンピュータの一つが会議モデル・データを処理する命令を生成し、サーバ・コンピュータへ前記命令を送るステップと、(c) 前記クライアント・コンピュータの各々がサーバ・コンピュータからの命令のコピーを受信し、該命令にもとづいて上記クライアント・コンピュータの各々のモデルを処理するステップとを有し、異なるクライアントからの複数の異なる処理命令の後に、前記会議モデル・データの各々のコピーは互に対応するパーソナル会議方法。

(3) 前記サーバ・コンピュータは、前記会議モデル・データを要求あり次第クライアント・コンピュータへ前記カンファレンス・モデル・データを送る上記(1)または(2)に記載のパーソナル会議方法。

(4) 上位マシン・バイト・コードの制御下で前記ステップが実行される上記(1)ないし(3)のいずれか一項に記載のパーソナル会議方法。

(5) クライアント・コンピュータを制御する上位マシン・バイト・コードは、該クライアント・コンピュータで実行されるネットワーク・ブラウザによって前記サーバ・コンピュータからダウンロード可能である上記

(4)に記載のパーソナル会議方法。

(6) 前記会議モデル・データを処理する前記命令は変数を有するOp codeの形態である上記(2)ないし

(5)のいずれか一項に記載のパーソナル会議方法。

(7) 会議モデル・データとクライアント・コンピュータから命令を受信して該命令にもとづいて前記会議モデル・データを処理する手段とを有するサーバ・コンピュータと、命令を生成して該命令を前記サーバ・コンピュータへ送る手段とを有する少なくとも2台のクライアント・コンピュータとを備えるパーソナル会議システムであって、前記サーバ・コンピュータは、受信した命令をすべてのクライアント・コンピュータへ分散する手段をさらに有し、また前記少なくとも2台のクライアント・コンピュータの各々は、会議モデル・データのコピーと、前記サーバ・コンピュータからの分散された命令を受信して前記サーバ・コンピュータから受信した命令にもとづいて前記会議モデル・データのコピーを処理する手段とをさらに有するパーソナル会議システム。

(8) 前記少なくとも2台のクライアント・コンピュータの各々は、前記会議モデル・データを表示するための手段をさらに有する上記(7)に記載のパーソナル会議システム。

(9) 会議モデル・データと、該会議モデル・データを処理するための命令を生成して該命令を他のコンピュータへ送る手段と、前記命令にもとづいて前記会議モデル・データを処理する手段とを有し、かつメッセージ通信のために接続した少なくとも2台のコンピュータを有するパーソナル会議システムであって前記少なくとも2台のコンピュータの一つがサーバ・コンピュータであり、他のコンピュータが該サーバ・コンピュータと接続されたクライアント・コンピュータであり、前記サーバ・コンピュータは、さらに接続されたクライアント・コンピュータのすべてに分散するための手段を有し、さらに、前記会議モデル・データを処理する手段は、前記サーバ・コンピュータから分散された命令を受信するための手段を有するもので、異なるコンピュータからの処理命令のシーケンスは前記会議モデル・データの各々のコピーに対して同時に実行されるパーソナル会議システム。

(10) サーバ・オブジェクトおよび複数のクライアント・オブジェクトと、モデル・データと命令メッセージに対して前記モデル・データを処理する手段とを有するモデル・オブジェクトとを有し、また前記複数のクライアント・オブジェクトの各々は、前記モデル・オブジェクトを処理するための命令を生成する手段を有するパーソナル会議システムにおいて、前記複数のクライアント・オブジェクトの各々は関連したモデル・オブジェクトを有し、また前記複数のクライアント・オブジェクトの各々は処理命令をサーバ・オブジェクトに送る手段をさらに有することによって、実行されるべき異なるクライアント・オブジェクトからの命令のシーケンスをモデル・データのインスタンス上の各モデル・オブジェクトによって同時に実行させるパーソナル会議システム。

(11) コンピュータ・ネットワーク全体にわたる伝送

のためにモデル・オブジェクトとクライアント・オブジェクトとを実装するための手段をさらに有する上記(10)に記載のパーソナル会議システム。

(12) 前記モデル・オブジェクトは、現在のモデル・データを含む状態メッセージを発生させ、該状態メッセージを他のオブジェクトに送るために、モデル・オブジェクトを生成する手段をさらに有する上記(11)に記載のパーソナル会議システム。

(13) 前記モデル・オブジェクトは、前記モデル状態メッセージに送られたものへモデル・データをセットし、それによって一つのモデル・オブジェクトが他のモデル・オブジェクトによってコピーされるために、前記モデル・オブジェクトを生成する手段をさらに有する上記(12)に記載のパーソナル会議システム。

(14) さらに、前記サーバ・オブジェクトに関連して集中化されたモデル・オブジェクトを有する上記(10)ないし(13)のいずれか一項に記載のパーソナル会議システム。

(15) クライアント・オブジェクトに関連したモデル・オブジェクトは前記サーバ・オブジェクトに関連した前記モデル・オブジェクトからコピーされる上記(14)に記載のパーソナル会議システム。

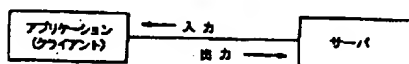
(16) 前記クラスおよびオブジェクトは、上位マシン・バイト・コードにインプリメントされる上記(10)ないし(15)のいずれか一項に記載のパーソナル会議システム。

(17) クライアント・クラスおよびモデル・クラスは、ダウンロード可能で、かつ前記上位マシン・バイト・コードによって使用可能となるネットワーク・ブラウザによって実行可能な上位マシン・バイト・コード・アプレットを含む上記(16)に記載のパーソナル会議システム。

#### 【図面の簡単な説明】

【図1】従来のピア・トゥ・ピア会議システムの一例を

【図2】



示すブロック図である。

【図2】従来のクライアント/サーバ会議システムの一例を示すブロック図であり、標準的なXWindowにおけるクライアントとサーバとのインタラクションを示す。

【図3】従来のクライアント/サーバ会議システムの一例を示すブロック図であり、クライアントとサーバとの間に設けられたXTVとのインタラクションを示す。

【図4】本発明にもとづくパーソナル会議システムの実施形態例を説明するためのブロック図である。

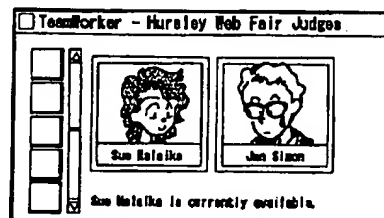
【図5】本発明にもとづくパーソナル会議システムをインプリメントする際に使用されるプログラム・クラスを模式的に説明するためのブロック図である。

【図6】本発明にもとづくパーソナル会議システムに適用されるIBMのTeam Workerアプリケーションのコンピュータ画面を表す図である。

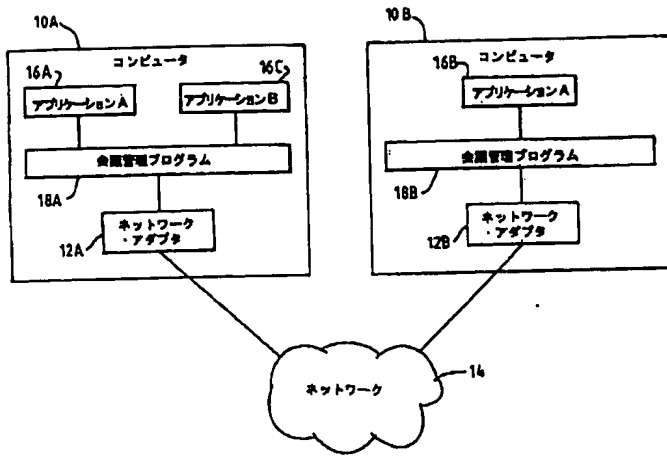
#### 【符号の説明】

|                |                   |
|----------------|-------------------|
| 30             | サーバ・コンピュータ        |
| 32 A、32 B、32 C | クライアント・コンピュータ     |
| 34             | プロセッサ             |
| 36 A、36 B、36 C | クライアント・コード        |
| 38             | メモリ               |
| 40             | 周辺機器              |
| 42             | キーボード             |
| 43             | マウス               |
| 44             | ネットワーク・アダプタ       |
| 47             | プロセッサ             |
| 48             | プロセッサ             |
| 50 A-E         | サーバ・コード           |
| 52             | メモリ               |
| 56             | クライアント・アプレット      |
| 58             | WWWのページ (ウェブ・ページ) |
| 60             | モデル・データ           |

【図6】



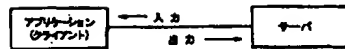
【図 1】



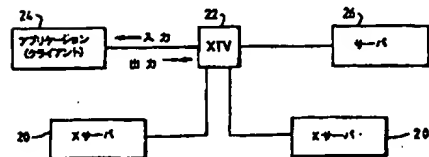
【図 3】

14k996063

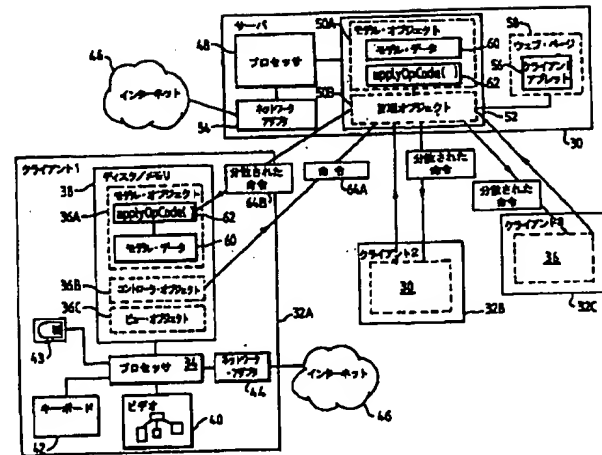
【図 2】



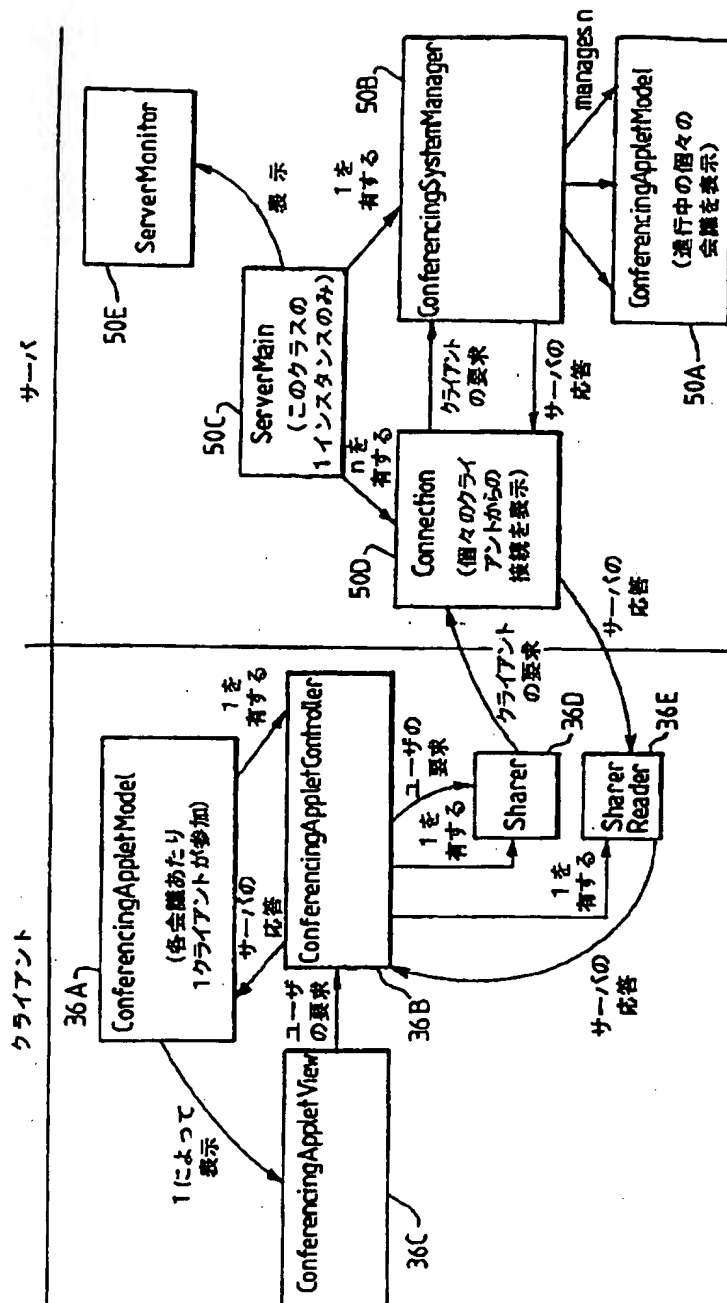
【図 3】



【図 4】



【図5】



【手続補正書】

【提出日】平成10年4月3日

【手続補正1】

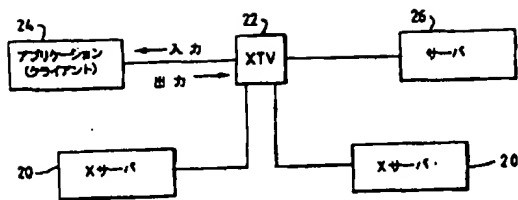
【補正対象書類名】図面

【補正対象項目名】図3

【補正方法】変更

【補正内容】

【図3】



フロントページの続き

(72)発明者 アンドリュー・ケイ  
イギリス国エスオー16 8 ジェイイー、  
ハンプシャー州サウスハンプトン ローン  
アムス レッドウォード・ロード 18

(72)発明者 ヴィンス・セスイ  
イギリス国エスオー16 8 エルティー、  
ハンプシャー州サウスハンプトン ローン  
アムス フリップス・クローズ 3

**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

**BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☒ FADED TEXT OR DRAWING
- ☒ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☒ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**